The parallelization basically works determining which slice it is in and passing all the relevant arguments from lifeparellel in from the original lifeseq method. Then by using the slice it will divide nrows into four equal slice and for each thread it will take i and only iterate across each of these slices. Since we know that for each iteration we choose a value of j as it iterates across the NxN matrix since each will be contiguous we therefore we can use previous value of north, south, northeast east, and southeast rather than recalculating all of it so this optimization will take all the previous values that have already been previously calculated and store them in temporary variables and update them as we move along j that way we only need to the read operation once rather than twice which occurs in the original lifeseq. Another optimization we can do is in-lining this seemed to improve the performance a lot we removed all the macros and method calls and place them into just embedded code since all the code is just one liners so we just replace the method call with that one line to inline each method and macro. These optimization were the main ones that were done. Grid tiling was also tried but it seemed to have a very minimal effect so it was not included int the final solution.